REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
27-09-2007	Technical Memo	

4. TITLE AND SUBTITLE

Particle Motion Algorithm for Arbitrary Gyro-Frequencies

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Jean-Luc Cambier (AFRL/RZSA); Oleg Batishchev (MIT-Space Systems

5d. PROJECT NUMBER

5e. TASK NUMBER
23080532

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory (AFMC)
AFRL/RZSA
10 E. Saturn Blvd.
Edwards AFB CA 93524-7680

8. PERFORMING ORGANIZATION REPORT NUMBER

AFRL-RZ-ED-TP-2007-431

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory (AFMC)
AFRL/RZS
5 Pollux Drive
Edwards AFB CA 93524-7048

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S NUMBER(S)
AFRL-RZ-ED-TP-2007-431

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution unlimited (PA #07376A).

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The transport of particles in a Particle-In-Cell (PIC) method is traditionally handled by a staggered algorithm, second-order accurate in time, originally developed by Boris [1-2]. The scheme is very efficient and although it is stable for time steps large compared to the cyclotron period ("gyro-period"), it ceases to be accurate in that case. In cases of strong applied magnetic field, this can impose an impractical time-step restriction. An alternative approach is to average over the orbital motion and consider only that of the guiding-center; this has led to so-called gyrokinetic simulations [3]. However, that approach can also lead to some inaccuracies, due to the loss of information regarding the phase of the orbital motion. Furthermore, it may also be desirable to have an algorithm that is not staggered in time, in order to guarantee exact conservation of total energy at all times. In this paper, we present an algorithm that solves the non-relativistic equation of motion exactly, and can yield exact conservation of energy for large time steps (compared to gyroperid). The algorithm accuracy is demonstrated and compared with the Boris scheme. These preliminary results are valid for the homogenous case only, and extension to spatially-varying fields should be considered next.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Dr. Jean-Luc Cambier
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	SAR	18	**19b. TELEPHONE NUMBER** *(include area code)*
Unclassified	**Unclassified**	**Unclassified**			N/A

Particle Motion Algorithm for Arbitrary Gyro-Frequencies

Jean-Luc Cambier
AFRL, Edwards AFB, CA 93524

Oleg Batishchev
MIT – Space Systems Laboratory, Cambridge, MA 02139

Abstract

The transport of particles in a Particle-In-Cell (PIC) method is traditionally handled by a staggered algorithm, second-order accurate in time, originally developed by Boris [1-2]. The scheme is very efficient and although it is stable for time steps large compared to the cyclotron period ("gyro-period"), it ceases to be accurate in that case. In cases of strong applied magnetic field, this can impose an impractical time-step restriction. An alternative approach is to average over the orbital motion and consider only that of the guiding-center; this has led to so-called gyrokinetic simulations [3]. However, that approach can also lead to some inaccuracies, due to the loss of information regarding the phase of the orbital motion. Furthermore, it may also be desirable to have an algorithm that is not staggered in time, in order to guarantee exact conservation of total energy at all times. In this paper, we present an algorithm that solves the non-relativistic equation of motion exactly, and can yield exact conservation of energy for large time steps (compared to gyroperid). The algorithm accuracy is demonstrated and compared with the Boris scheme. These preliminary results are valid for the homogenous case only, and extension to spatially-varying fields should be considered next.

1. Introduction

We consider the problem of solving the non-relativistic dynamical equation for charged particles in arbitrary electric and magnetic fields:

$$m\dot{\vec{v}} = q\,(\vec{E} + \vec{v} \times \vec{B}) \tag{1}$$

This is the basic transport process in Particle-In-Cell (PIC) codes, which is usually solved using the Boris algorithm [1], defined in Appendix A. The Boris algorithm is a computationally efficient (i.e. uses a minimum number of operations) algorithm, second-order accurate in time. Since it is a leap-frog integrator, it is also usually described as a symplectic algorithm, i.e. which conserves a discrete analog of the Hamiltonian up to second-order accuracy. This is a critically important property for PIC simulations, which usually do not have conservation properties embedded in the mathematical formulation as in continuum models, such as finite-volume or finite-difference schemes. However, it is important to exercise some caution when speaking of energy conservation in the Boris scheme; as a leap-frog algorithm, it uses position and velocity staggered in time, the kinetic and potential energies are not computed at the same time. After advancing the particle, the kinetic energy can be evaluated from the velocity field at time $(n+1/2)$, while the potential energy can be obtained exactly from the particle position at time $(n+1)$. Thus, the kinetic and potential energies are not strictly conserved at the same time.

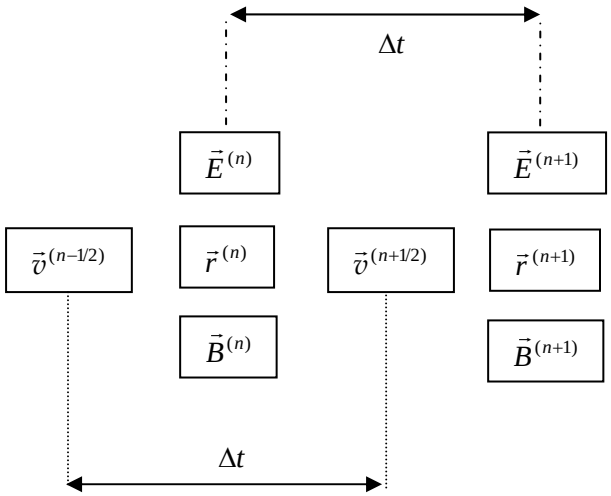

Figure 1: Schematic of leap-frog Boris algorithm.

In the leap-frog algorithm, the fields are used at the mid-point for advancing the velocity, i.e. fields evaluated at time (n) are needed for updating the velocity from $\vec{v}^{(n-1/2)}$ to $\vec{v}^{(n+1/2)}$. In an electrostatic simulation, the electric field can be obtained from solving Poisson's equation:

$$\vec{\nabla}^2 \phi = \frac{e}{\varepsilon_0}(Z_i \bar{n}_i - \bar{n}_e) \tag{2a}$$

and

$$\vec{E}^{(n)} = -\vec{\nabla}\phi^{(n)} \tag{2b}$$

In (2a), the particle density at a given location (grid-point) is obtained as the statistical average of the contribution of neighboring particle; this "scatter" operation maps the particles onto the grid, and various interpolation schemes can be used for this operation. We point out that this mapping

uses the particle locations at time level (n), and therefore the electro-static potential (and electric field) are naturally synchronized with the particle positions. In electro-static simulations the magnetic field is constant and there is no concern over its synchronization. In electro-magnetic simulations, however, both fields are advanced in time and the procedure must be consistent with the Maxwell equations:

$$\varepsilon_0 \vec{\nabla} \cdot \vec{E} = -\tilde{\rho}_p \tag{3a}$$

$$\varepsilon_0 \frac{\partial \vec{E}}{\partial t} = -\frac{\vec{\nabla} \times \vec{B}}{\mu_0} + \vec{j}_p \tag{3b}$$

$$\frac{\partial \vec{B}}{\partial t} = -\vec{\nabla} \times \vec{E} \tag{3c}$$

$$\vec{\nabla} \cdot \vec{B} = 0 \tag{3d}$$

Equation (3a), where $\tilde{\rho}_p$ is the charge density from the particles, is simply Poisson's equation (2a); in (3b), $\vec{j}_p$ is the current density from the particles, and is obtained by a similar mapping of the particle velocities onto grid points. The Maxwell equations are naturally synchronized to second-order accuracy for $\vec{r}^{(n)}, \vec{E}^{(n)}$ and $\vec{v}^{(n+1/2)}, \vec{B}^{(n+1/2)}$. However the leap-frog algorithm for particle transport is of the form $\vec{v}^{(n+1/2)} = f(\vec{v}^{(n-1/2)}, \vec{E}^{(n)}, \vec{B}^{(n)})$, and one needs to interpolate in time one of the fields for the particle push, i.e. $\vec{B}$. Note that the leap-frog algorithm of Figure 1 is not the unique solution: one could just as well decide to choose the fields $\vec{r}^{(n)}, \vec{B}^{(n)}$ and $\vec{v}^{(n+1/2)}, \vec{E}^{(n+1/2)}$ or other combinations and rely on the time-interpolation of another field (matter or particle) to re-establish second-order time accuracy. Higher-order schemes can of course be obtained with iterative methods.

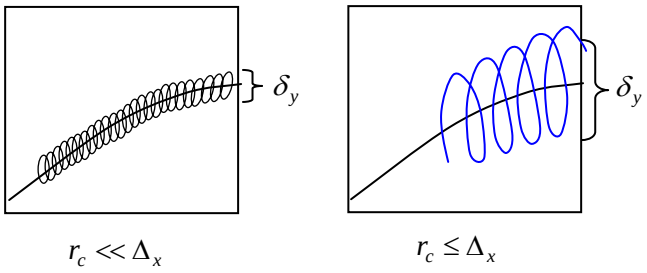

Figure 2: Potential positional error of drift dynamics versus gyro-radius

The Boris algorithm is stable at high values of the magnetic field, i.e. when $\omega_c \Delta t >> 1$, although accuracy is lost for large time steps. Practically speaking, the time step in PIC simulations using this algorithm is restricted such that $\omega_c \Delta t << 1$; this makes the scheme highly inefficient in cases of strongly magnetized plasmas. One could consider an alternative approach in that case, where only the motion of the guiding center is modeled; the rotation around the field line is not tracked, but averaged over several orbits. This "drift dynamics" approach is valid when the gyro-radius $r_c << \Delta_x$, the characteristic cell-size; however, significant errors can be introduced even when $r_c \leq \Delta_x$. Since the phase of the gyro-motion is not known in this approximation, the particle position is effectively randomized on a scale comparable to the cell size (see Figure 2). This can lead to errors at the crossing into different cells or boundaries, and errors when the field gradients on the scale of a cell size are non-negligible. Therefore, it is worth investigating the construction

of an accurate particle-push algorithm that is more efficient at high cyclotron frequency, yet remains accurate and conserves energy to a high level of accuracy. This is the object of the following study.

2. Exact Solution

We consider here the case of constant and uniform fields. This considerably simplifies the analysis and allows us to obtain an exact analytical solution to the non-relativistic equations of motion. The constant field approximation is valid when the time-variation of the fields is neglected during the time-step (i.e. first-order time-accuracy of the field evolution); the extension to higher-order time-dependency and non-uniform fields will be examined in the future. We will also be performing a transformation to the reference frame aligned with the magnetic field. Let us first define the laboratory frame (L) by the italicized letters $(\hat{x},\hat{y},\hat{z})$ and a rotated coordinate frame by $(\hat{\xi},\hat{\eta},\hat{\zeta})$ such that the unit vector $\hat{\zeta}$ is aligned with the magnetic field, i.e. $\hat{\zeta}\equiv\hat{b}$.

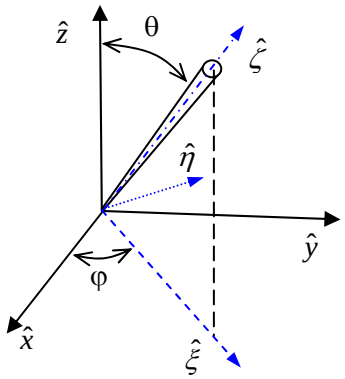

Figure 3: Reference frame transformation: $\hat{\zeta}$ aligned with $\vec{B}$.

The rotation operators between the two reference frames are given by:

$$\hat{\mathbf{R}}=\begin{pmatrix}\xi_x & \xi_y & \xi_z\\ \eta_x & \eta_y & \eta_z\\ \zeta_x & \zeta_y & \zeta_z\end{pmatrix}=\begin{pmatrix}c_\varphi c_\theta & s_\varphi c_\theta & -s_\theta\\ -s_\varphi & c_\varphi & 0\\ c_\varphi s_\theta & s_\varphi s_\theta & c_\theta\end{pmatrix} \tag{4a}$$

$$\hat{\mathbf{R}}^{-1}=\begin{pmatrix}\xi_x & \eta_x & \zeta_x\\ \xi_y & \eta_y & \zeta_y\\ \xi_z & \eta_z & \zeta_z\end{pmatrix}=\begin{pmatrix}c_\varphi c_\theta & -s_\varphi & c_\varphi s_\theta\\ s_\varphi c_\theta & c_\varphi & s_\varphi s_\theta\\ -s_\theta & 0 & c_\theta\end{pmatrix} \tag{4b}$$

where we have used the condensed notation of $c_\varphi=\cos(\varphi)$, $s_\theta=\sin(\theta)$, etc. Once the rotation into the aligned frame is performed and no confusion is possible, we can use the script letters $(\mathrm{x},\mathrm{y},\mathrm{z})$ to denote the components in that frame, i.e. $(\hat{\mathrm{x}},\hat{\mathrm{y}},\hat{\mathrm{z}})\equiv(\hat{\xi},\hat{\eta},\hat{\zeta})$. We will also denote vectors in that frame by bold-face type, i.e.: $\mathbf{E}=\hat{\mathbf{R}}\otimes\vec{E}$.

In this rotated frame, the equation of motion (1) can be expressed by:

$$\begin{aligned}\dot{\mathrm{v}}_\mathrm{x} &= \frac{q}{m}\mathrm{E}_\mathrm{x} + \frac{qB}{m}\mathrm{v}_\mathrm{y}\\ \dot{\mathrm{v}}_\mathrm{y} &= \frac{q}{m}\mathrm{E}_\mathrm{y} - \frac{qB}{m}\mathrm{v}_\mathrm{x}\\ \dot{\mathrm{v}}_\mathrm{z} &= \frac{q}{m}\mathrm{E}_\mathrm{z}\end{aligned} \tag{5}$$

In this frame, the magnetic field has a component only in the $\hat{\mathrm{z}}$ direction, and therefore $\mathrm{B}_\mathrm{z} \equiv B$, the magnitude of the magnetic field. We can define a normalized electric acceleration field $\mathbf{a} = q\mathbf{E}/m$ and the cyclotron frequency $\omega = qB/m$, which is a *signed* quantity. The solution for the $\hat{\mathrm{z}}$-component of the velocity is trivial and can be ignored for the moment. The system (5) can be reduced, for the transverse components, to:

$$\dot{\underline{\mathrm{v}}} = \underline{\mathrm{a}} + \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} \cdot \underline{\mathrm{v}} \tag{6}$$

where the underline indicates a vector in the transverse directions only. An additional time derivative of (6) yields the following:

$$\ddot{\underline{\mathrm{v}}} = -\omega^2 \underline{\mathrm{v}} + \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} \cdot \underline{\mathrm{a}} \tag{7}$$

The general solution of (6-7) is:

$$\underline{\mathrm{v}} = \begin{cases} \mathrm{v}_\mathrm{o}\sin(\omega t{+}\varphi) + (\mathrm{a}_\mathrm{x}/\omega)\sin(\omega t{+}\varphi) + (\mathrm{a}_\mathrm{y}/\omega)\big(1{-}\cos(\omega t{+}\varphi)\big) \\ \mathrm{v}_\mathrm{o}\cos(\omega t{+}\varphi) - (\mathrm{a}_\mathrm{x}/\omega)\big(1{-}\cos(\omega t{+}\varphi)\big) + (\mathrm{a}_\mathrm{y}/\omega)\sin(\omega t{+}\varphi) \end{cases} \tag{8}$$

Let us denote $\phi(t) = \omega t + \varphi$. It is to verify that the solution (8) satisfies the equations of motion:

$$\dot{\underline{\mathrm{v}}} = \begin{cases} \omega\mathrm{v}_\mathrm{o}\cos(\phi) + \mathrm{a}_\mathrm{x}\cos(\phi) + \mathrm{a}_\mathrm{y}\sin(\phi) \\ -\omega\mathrm{v}_\mathrm{o}\sin(\phi) - \mathrm{a}_\mathrm{x}\sin(\phi) + \mathrm{a}_\mathrm{y}\cos(\phi) \end{cases} = \begin{cases} \omega\mathrm{v}_\mathrm{y} + \mathrm{a}_x \\ -\omega\mathrm{v}_\mathrm{x} + \mathrm{a}_y \end{cases} = \underline{\mathrm{a}} + \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} \cdot \underline{\mathrm{v}} \quad \boxtimes \text{ (9a)}$$

$$\ddot{\underline{\mathrm{v}}} = \begin{cases} -\omega^2\mathrm{v}_\mathrm{o}\sin(\phi) - \mathrm{a}_\mathrm{x}\omega\sin(\phi) + \mathrm{a}_\mathrm{y}\omega\cos(\phi) \\ -\omega^2\mathrm{v}_\mathrm{o}\cos(\phi) - \mathrm{a}_\mathrm{x}\omega\cos(\phi) - \mathrm{a}_\mathrm{y}\omega\sin(\phi) \end{cases} = \begin{cases} -\omega^2\mathrm{v}_\mathrm{x} + \omega\mathrm{a}_\mathrm{y} \\ -\omega^2\mathrm{v}_\mathrm{y} - \omega\mathrm{a}_\mathrm{x} \end{cases} = -\omega^2\underline{\mathrm{v}} + \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} \cdot \underline{\mathrm{a}} \quad \boxtimes \text{ (9b)}$$

Let us now compute the solution at an advanced time $t + dt$. From (8) we have:

$$\underline{\mathrm{v}}(t{+}dt) = \begin{cases} \mathrm{v}_\mathrm{o}\sin(\phi{+}d\phi) + (\mathrm{a}_x/\omega)\sin(\phi{+}d\phi) + (\mathrm{a}_y/\omega)\big(1{-}\cos(\phi{+}d\phi)\big) \\ \mathrm{v}_\mathrm{o}\cos(\phi{+}d\phi) - (\mathrm{a}_x/\omega)\big(1{-}\cos(\phi{+}d\phi)\big) + (\mathrm{a}_y/\omega)\sin(\phi{+}d\phi) \end{cases} \tag{10}$$

Expanding the trigonometric functions we find:

$$\underline{\mathrm{v}}(t{+}dt) = \begin{cases} c_\delta \cdot \big[\mathrm{v}_\mathrm{x} - \mathrm{a}_\mathrm{y}/\omega\big] + s_\delta \cdot \big[\mathrm{v}_\mathrm{y} + \mathrm{a}_\mathrm{x}/\omega\big] + (\mathrm{a}_\mathrm{y}/\omega) \\ -s_\delta \cdot \big[\mathrm{v}_\mathrm{x} - \mathrm{a}_\mathrm{y}/\omega\big] + c_\delta \cdot \big[\mathrm{v}_\mathrm{y} + \mathrm{a}_\mathrm{x}/\omega\big] - (\mathrm{a}_\mathrm{x}/\omega) \end{cases} = \underline{\boldsymbol{\Omega}}(dt) \cdot \underline{\mathrm{v}}(t) + \frac{1}{\omega}\begin{bmatrix} s_\delta & 1{-}c_\delta \\ c_\delta{-}1 & s_\delta \end{bmatrix} \cdot \underline{\mathrm{a}} \tag{11}$$

where $\delta \equiv \omega\, dt$ and $\underline{\boldsymbol{\Omega}}(dt)$ is the counter-rotation matrix around the magnetic field:

$$\underline{\boldsymbol{\Omega}}(dt) = \begin{pmatrix} \cos(\delta) & \sin(\delta) \\ -\sin(\delta) & \cos(\delta) \end{pmatrix} \tag{12}$$

The last matrix in (11) can also be written in terms of this rotation matrix. Let us define:

$$\underline{\boldsymbol{\sigma}} = \omega\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{and} \quad \underline{\boldsymbol{\sigma}}^{-1} = \omega^{-1}\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \tag{13}$$

then (11) becomes:

$$\underline{\mathrm{v}}(t{+}\Delta t) = \underline{\boldsymbol{\Omega}}(dt) \cdot \underline{\mathrm{v}}(t) + \begin{bmatrix} c_\delta{-}1 & s_\delta \\ -s_\delta & c_\delta{-}1 \end{bmatrix} \cdot \boldsymbol{\sigma}^{-1} \cdot \underline{\mathrm{a}} \tag{14}$$

In a compact form:

$$\Delta\underline{\mathrm{v}} = \underline{\mathrm{v}}(t+\Delta t)-\underline{\mathrm{v}}(t) = \left[\underline{\boldsymbol{\Omega}}(dt) - \underline{\mathbf{1}}\right]\cdot\left[\underline{\mathrm{v}}(t)+\boldsymbol{\sigma}^{-1}\cdot\underline{\mathrm{a}}\right] \tag{15}$$

One should now consider the case of vanishing magnetic field. The matrix $\underline{\boldsymbol{\sigma}}^{-1}$ on the RHS of (15) is singular when $B=0$. However, it can be combined with the term in brackets as follows:

$$\left[\underline{\boldsymbol{\Omega}}(dt) - \underline{\mathbf{1}}\right]\cdot\underline{\boldsymbol{\sigma}}^{-1} = \underbrace{\begin{pmatrix} \dfrac{\sin\delta}{\delta} & \dfrac{1-\cos\delta}{\delta} \\ \dfrac{\cos\delta-1}{\delta} & \dfrac{\sin\delta}{\delta} \end{pmatrix}}_{\underline{\boldsymbol{\Delta}}_1}\cdot dt \tag{16}$$

The matrix $\underline{\boldsymbol{\Delta}}_1$ is regular, since:

$$\underline{\boldsymbol{\Delta}}_1 \to \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{when} \quad \omega \to 0 \tag{17}$$

One can expand (15) to the next order in $\delta = \omega\Delta t$, leading to:

$$\Delta\underline{\mathrm{v}} = \begin{bmatrix} \mathrm{a}_x dt+\omega\, dt(\mathrm{v}_y+\frac{1}{2}\mathrm{a}_y dt) - \frac{1}{2}(\omega\, dt)^2\,\mathrm{v}_x \\ \mathrm{a}_y dt-\omega\, dt(\mathrm{v}_x+\frac{1}{2}\mathrm{a}_x dt) - \frac{1}{2}(\omega\, dt)^2\,\mathrm{v}_y \end{bmatrix} \tag{18}$$

We see that the second-order accurate ($o(dt^2)$) solution is obtained by rotating a half-step advanced solution, as expected. We can also verify that this solution is identical to the Boris algorithm, by comparing (18) with (A.23b) of Appendix A.

The opposite limit of large time steps compared to the gyro-motion, i.e. $\omega\, dt \to \infty$, is also of principal interest. In that case, the trigonometric functions oscillate rapidly, but the trajectory remains bound. One can perform an averaging over a large number of gyro-motions, and eliminate all terms proportional to these functions ($<\cos>=<\sin>=0$). The remainder is:

$$<\underline{\mathrm{v}}(t+\Delta t)>=\begin{pmatrix} 0 & 1/\omega \\ -1/\omega & 0 \end{pmatrix}\cdot\underline{\mathrm{a}} \tag{19}$$

which is <u>independent of the time step</u> Δt. This is a constant velocity, which can be easily recognized as the $\vec{E}\times\vec{B}$ drift velocity, since (19) is equivalent to:

$$<\underline{\mathrm{v}}>=\begin{cases} E_y / B \\ -E_x/B \end{cases} = \frac{\mathbf{E}\times\mathbf{B}}{B^2} \tag{20}$$

Therefore, the formulation (15) automatically recovers the drift motion of the guiding center when the gyro-motion is not resolved – with a randomized rotation around the magnetic field.

Let us now look at the exact solution for the particle position. From (8), we obtain:

$$\underline{\mathrm{x}}(t) = \frac{\mathrm{v}_o}{\omega}\begin{pmatrix} -\cos\phi \\ \sin\phi \end{pmatrix} + \frac{1}{\omega^2}\begin{pmatrix} -\cos\phi & -\sin\phi \\ \sin\phi & -\cos\phi \end{pmatrix}\cdot\underline{\mathrm{a}} + \frac{1}{\omega}\begin{pmatrix} \mathrm{a}_y t \\ -\mathrm{a}_x t \end{pmatrix} + \underline{\mathrm{x}}_0 \tag{21}$$

The expression at a later time $t+dt$ can be expressed as function of the original phase $\phi(t)$ and the phase difference $d\phi \equiv \delta = \omega\, dt$ by:

$$\underline{\mathrm{x}}(t+dt) = \frac{1}{\omega}\begin{pmatrix} \mathrm{a}_y(t+dt) \\ -\mathrm{a}_x(t+dt) \end{pmatrix} + \underline{\mathrm{x}}_0 + \begin{pmatrix} c_\delta & s_\delta \\ -s_\delta & c_\delta \end{pmatrix}\cdot\begin{bmatrix} -(\mathrm{v}_o/\omega)\,c_\phi & -(\mathrm{a}_x/\omega^2)c_\phi & -(\mathrm{a}_y/\omega^2)s_\phi \\ +(\mathrm{v}_o/\omega)\,s_\phi & +(\mathrm{a}_x/\omega^2)s_\phi & -(\mathrm{a}_y/\omega^2)c_\phi \end{bmatrix} \tag{22}$$

One recognizes again the rotation matrix (12) in that expression, which allows us to write the displacement as:

$$\Delta\underline{\mathrm{x}} = \frac{1}{\omega}\begin{pmatrix} \mathrm{a}_y dt \\ -\mathrm{a}_x dt \end{pmatrix} + \left[\underline{\boldsymbol{\Omega}}(dt) - \underline{\mathbf{1}}\right]\cdot\begin{bmatrix} -(\mathrm{v}_o/\omega)\,c_\phi & -(\mathrm{a}_x/\omega^2)c_\phi & -(\mathrm{a}_y/\omega^2)s_\phi \\ +(\mathrm{v}_o/\omega)\,s_\phi & +(\mathrm{a}_x/\omega^2)s_\phi & -(\mathrm{a}_y/\omega^2)c_\phi \end{bmatrix} \tag{23}$$

However, from (15) one can also recognize the following expression:

$$\Delta\underline{\mathrm{v}} \equiv \left(\underline{\underline{\boldsymbol{\Omega}}}(dt) - \underline{\underline{\mathbf{1}}}\right) \cdot \begin{bmatrix} \mathrm{v}_x - \mathrm{a}_y/\omega \\ \mathrm{v}_y + \mathrm{a}_x/\omega \end{bmatrix} = \left(\underline{\underline{\boldsymbol{\Omega}}} - \underline{\underline{\mathbf{1}}}\right) \cdot \begin{bmatrix} \mathrm{v}_o s_\phi + (\mathrm{a}_x/\omega) s_\phi - (\mathrm{a}_y/\omega) c_\phi \\ \mathrm{v}_o c_\phi + (\mathrm{a}_x/\omega) c_\phi + (\mathrm{a}_y/\omega) s_\phi \end{bmatrix} \tag{24}$$

Inserting (24) into (23), we finally obtain:

$$\Delta\underline{\mathrm{x}} \equiv \frac{1}{\omega} \cdot \begin{bmatrix} \mathrm{a}_y \Delta t - \Delta \mathrm{v}_y \\ -\mathrm{a}_x \Delta t + \Delta \mathrm{v}_x \end{bmatrix} \tag{25}$$

For the displacement along the magnetic field, the exact solution is of course:

$$\Delta \mathrm{x}_{\parallel} = \mathrm{v}_{\parallel}(t)\,\Delta t + \tfrac{1}{2}\mathrm{a}_{\parallel}\Delta t^2 \equiv \mathrm{v}_{\parallel}(t + \tfrac{1}{2}\Delta t)\,\Delta t \tag{26}$$

It would appear that the transverse displacement (25) has a singular behavior at vanishing magnetic field strength, due to the ω^{-1} factor. However, a simple Taylor expansion can confirm that this is not the case: when $\omega \to 0$ one can use (18) into (25) to verify that, as expected:

$$\Delta\underline{\mathrm{x}} \approx \begin{bmatrix} \mathrm{v}_x \Delta t + \tfrac{1}{2}\mathrm{a}_x \Delta t^2 \\ \mathrm{v}_y \Delta t + \tfrac{1}{2}\mathrm{a}_y \Delta t^2 \end{bmatrix} + o(\omega^2) \tag{27}$$

The expression (25) is therefore valid for all non-zero values of the magnetic field. We can, as before, regularize this expression in the case of $\omega = 0$; after some simple algebra, we obtain:

$$\Delta\underline{\mathrm{x}} = \underbrace{\begin{pmatrix} \frac{s_\delta}{\delta} & \frac{1-c_\delta}{\delta} \\ \frac{c_\delta - 1}{\delta} & \frac{s_\delta}{\delta} \end{pmatrix}}_{\boldsymbol{\Delta}_1} \cdot \underline{\mathrm{v}}\,\Delta t + \underbrace{\begin{pmatrix} \frac{1-c_\delta}{\delta^2} & \frac{\delta - s_\delta}{\delta^2} \\ \frac{s_\delta - \delta}{\delta^2} & \frac{1-c_\delta}{\delta^2} \end{pmatrix}}_{\boldsymbol{\Delta}_2} \cdot \underline{\mathrm{a}}\,\Delta t^2 \tag{28}$$

We have recognized the first matrix (16), and defined a second regularized matrix $\underline{\underline{\boldsymbol{\Delta}}}_{\mathbf{2}}$. Both are finite when $\omega \to 0$, since in that case (defining also the following S, C coefficients):

$$S_0 = s_\delta \qquad S_1 \equiv \frac{s_\delta}{\delta} \approx 1 - \frac{\delta^2}{6} \qquad S_2 \equiv \frac{s_\delta - \delta}{\delta^2} \approx -\frac{\delta}{6} \tag{29a}$$

$$C_0 = 1 - c_\delta \qquad C_1 \equiv \frac{1-c_\delta}{\delta} \approx \frac{\delta}{2} \qquad C_2 \equiv \frac{1-c_\delta}{\delta^2} \approx \frac{1}{2} \tag{29b}$$

Therefore, the procedure outlined above is applicable in all cases of magnetic field values.

3. General Algorithm

One can construct two types of algorithms. The first case is valid only for $\delta > \varepsilon$, i.e. does not require regularization, and is governed by the following operations:

(1) Transform the velocity, position and acceleration vectors from the original reference frame into the rotated frame with the $\hat{\mathrm{z}}$ axis aligned with the magnetic field.

(2) Compute the changes to the transformed velocity vector, separating the transverse and parallel components:

$$\Delta\underline{\mathrm{v}} \equiv \left(\underline{\underline{\boldsymbol{\Omega}}}(dt) - \underline{\underline{\mathbf{1}}}\right) \cdot \begin{bmatrix} \mathrm{v}_x - \mathrm{a}_y/\omega \\ \mathrm{v}_y + \mathrm{a}_x/\omega \end{bmatrix} \tag{30a}$$

$$\Delta \mathrm{v}_{\parallel} = \mathrm{a}_{\parallel}\Delta t \tag{30b}$$

(3) Compute the changes to the transformed position, separating the transverse and parallel components:

$$\Delta\underline{\mathrm{x}} \equiv \frac{1}{\omega} \cdot \begin{bmatrix} \mathrm{a}_y \Delta t - \Delta \mathrm{v}_y \\ -\mathrm{a}_x \Delta t + \Delta \mathrm{v}_x \end{bmatrix} \tag{31a}$$

$$\Delta \mathrm{x}_{\parallel} = \mathrm{v}_{\parallel}(t)\,\Delta t + \tfrac{1}{2}\mathrm{a}_{\parallel}\Delta t^{2} \tag{31b}$$

(4) Transform the changes back into the original frame and add to the initial values.

In the second case, the regularized matrices are used so that the algorithm remains valid for all cases of field values, including $\delta < \varepsilon$. Combining transverse and parallel components*, we can express the velocity change as:

$$\Delta \mathbf{v} \equiv \underbrace{\begin{pmatrix} -C_0 & S_0 & 0 \\ -S_0 & -C_0 & 0 \\ 0 & 0 & 0 \end{pmatrix}}_{\mathbf{\Delta_0}} \cdot \begin{bmatrix} \mathrm{v}_x \\ \mathrm{v}_y \\ \mathrm{v}_z \end{bmatrix} + \underbrace{\begin{pmatrix} S_1 & C_1 & 0 \\ -C_1 & S_1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{\Delta_1}} \cdot \begin{bmatrix} \mathrm{a}_x \Delta t \\ \mathrm{a}_y \Delta t \\ \mathrm{a}_z \Delta t \end{bmatrix} \tag{32a}$$

$$\Delta \mathbf{x} = \underbrace{\begin{pmatrix} S_1 & C_1 & 0 \\ -C_1 & S_1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{\Delta_1}} \cdot \begin{bmatrix} \mathrm{v}_x \Delta t \\ \mathrm{v}_y \Delta t \\ \mathrm{v}_z \Delta t \end{bmatrix} + \underbrace{\begin{pmatrix} C_2 & -S_2 & 0 \\ S_2 & C_2 & 0 \\ 0 & 0 & 1/2 \end{pmatrix}}_{\mathbf{\Delta_2}} \cdot \begin{bmatrix} \mathrm{a}_x \Delta t^2 \\ \mathrm{a}_y \Delta t^2 \\ \mathrm{a}_z \Delta t^2 \end{bmatrix} \tag{32b}$$

The transformation steps (1) and (4) remain the same.

It would appear that the proposed scheme is very expensive, since it requires the evaluation of several matrices. However, in the case of constant fields studied so far, these matrices can be determined once the fields and time step are known. The transformation matrices (4) can be incorporated into the definition of the regularized push matrices $\mathbf{\Delta_0}, \mathbf{\Delta_1}, \mathbf{\Delta_2}$, leading to:

$$\mathbf{D_k} = \hat{\mathbf{R}}^{-1} \cdot \mathbf{\Delta_k} \cdot \hat{\mathbf{R}}, \quad k = 1,2,3 \tag{33}$$

The changes can then be computed directly in the initial (non-rotated) frame:

$$\Delta \vec{v} = \mathbf{D_0} \cdot \vec{v} + \frac{q\Delta t}{m} \mathbf{D_1} \cdot \vec{E} \tag{34a}$$

and

$$\Delta \vec{r} = \mathbf{D_1} \cdot \vec{v}\Delta t + \frac{q\Delta t^{2}}{m} \mathbf{D_2} \cdot \vec{E} \tag{34b}$$

4. Computational Tests

The first test conducted concerns the movement of a single particle (a positron) in static fields; the initial velocity is null, the electric field is 1 kV/m in the positive $\hat{y}$-direction and the magnetic field is 1 Tesla in the positive $\hat{z}$-direction. Under such initial conditions the particle executes a cycloidic movement of height equal to $h = 2r_L = 2\mathrm{v}_\mathrm{D}/\omega_c$, where $\mathrm{v}_\mathrm{D} = E/B$ is the drift velocity in the $\hat{x}$-direction. The motion is computed for three cases of constant time steps, being respectively $0.1/\omega_c$, $1/\omega_c$ and $10/\omega_c$. Since the Boris algorithm requires the velocity at a prior half-time step, that initial value ($\vec{v}^{(-1/2)}$) is computed from the exact solution. The trajectories for the exact solution, the Boris algorithm, and the regularized algorithm of eqs. (34) are shown in Figure 4. All methods are in very good agreement for small time step. For $\Delta t = 1/\omega_c$, the Boris algorithm starts to show some noticeable deviations from the exact solution; first, the Larmor radius, or height of the cycloid, is noticeably larger; second, the effective gyro-frequency is somewhat lower, leading to a growing de-phasing with the exact solution. At larger time steps (Figure 4c), the solution from the Boris algorithm is in error by close to an order of magnitude†. By contrast, the current algorithm provides a solution that is in perfect agreement with the exact solution, both in amplitude and phase.

---

* The underline is eliminated, since these are now 3-dimensional variables.

† Note that the magnitude of this error is bounded, i.e. does not grow in time, a result of symplecticity.

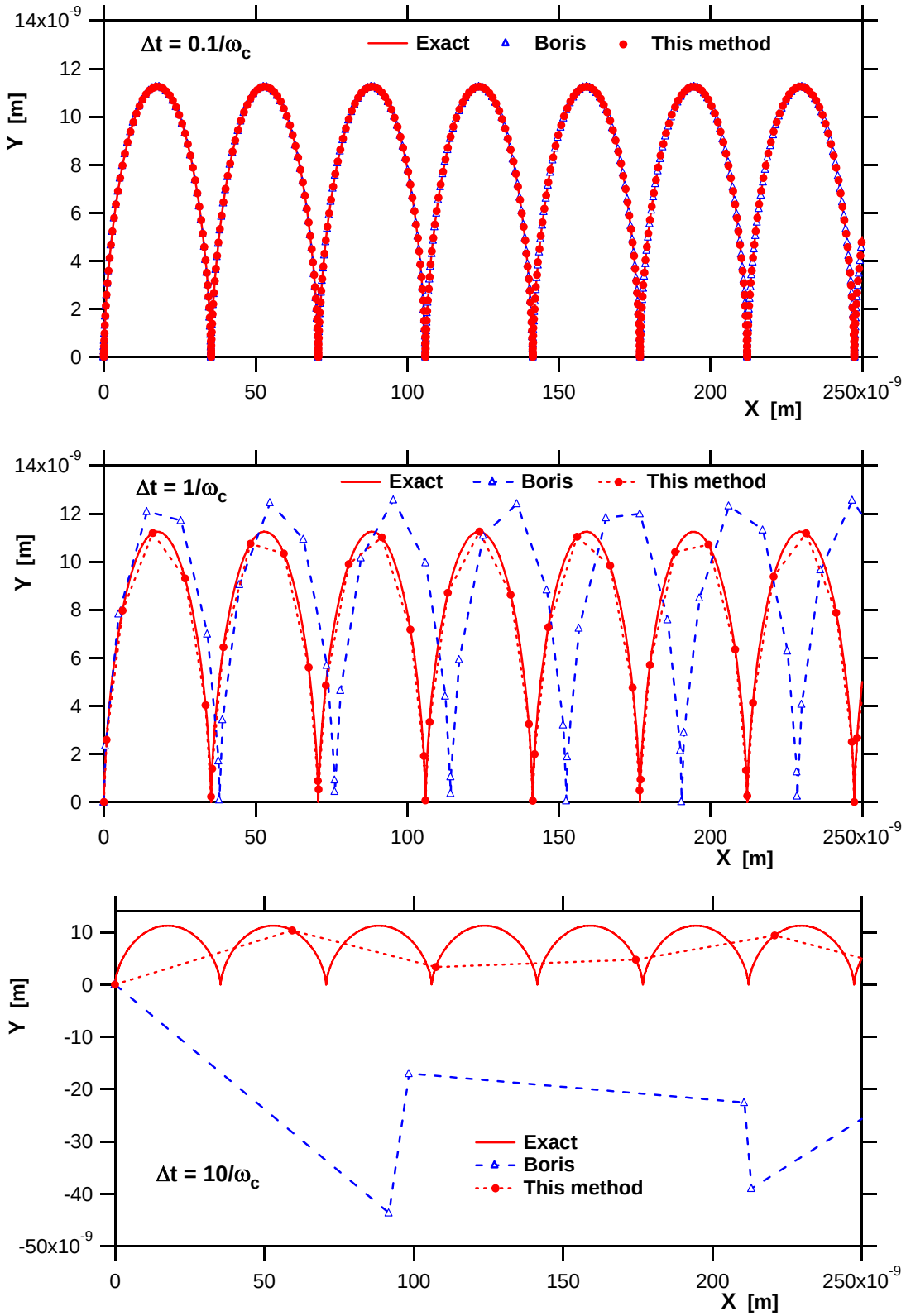

Figure 4: Particle trajectory for three time steps; exact solution is compared to the results from the Boris algorithm and the current scheme.

It should also be pointed out that, despite the obvious error in transverse position and rotation frequency present in the Boris algorithm, the drifting motion in the $\hat{x}$-direction is accurately maintained. This is evident in Figure 5, where the difference between the actual position and the expected position from the constant drift is plotted versus time. The natural oscillations are due to the cycloid motion itself and of amplitude equal to the Larmor radius; only at the largest time step does the Boris algorithm deviates from the expected behavior. The current method (Figure 5b) yields the correct drift dynamics at all time steps.

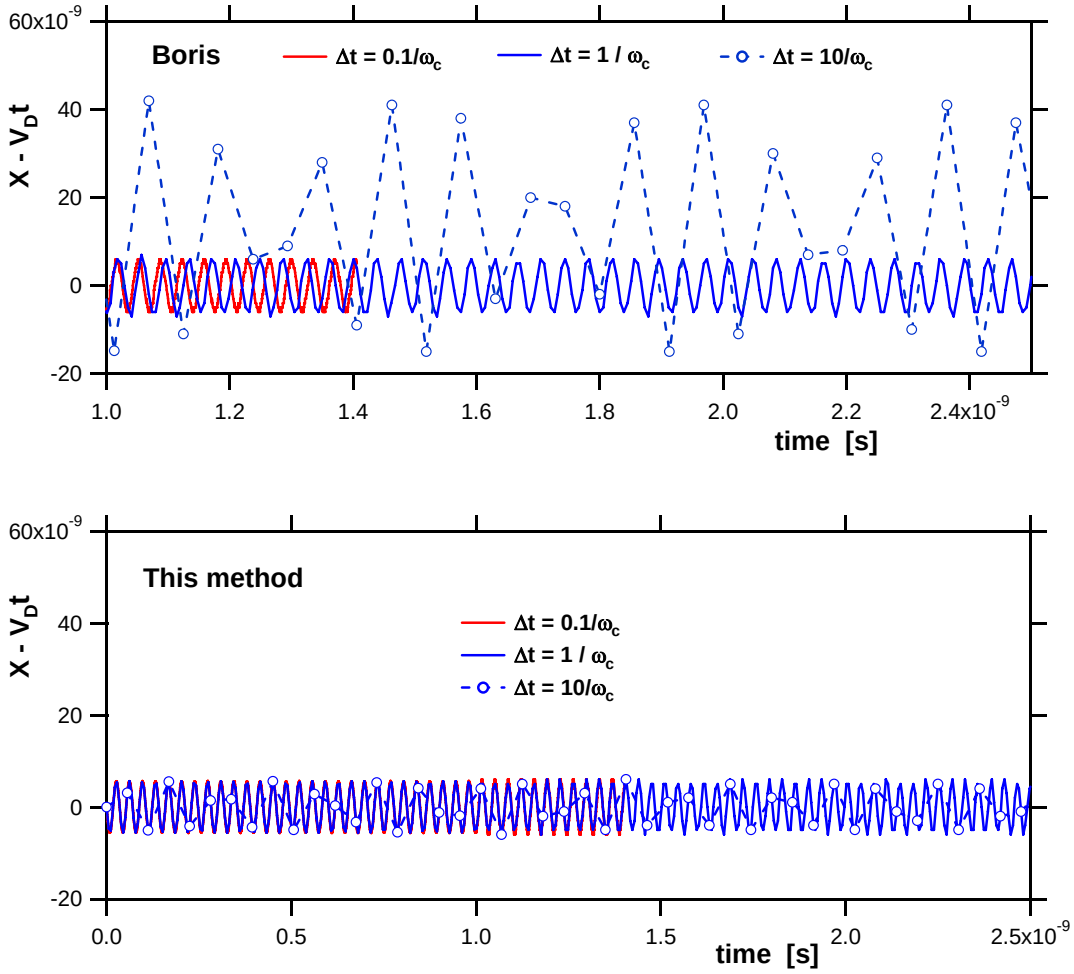

Figure 5: X-position versus time, normalized to theoretical position of guiding center ($V_D t$).

One can now examine the impact of the errors on conservation properties, i.e. kinetic, potential and total energies. In this simple test case with an imposed external field, the potential energy per mass is simply $e_{pot} = (qE/m) \cdot \mathrm{y}$, where y is the particle position along the $\hat{y}$-axis. It is important to point out that for the Boris algorithm, the kinetic and potential energies are evaluated at different times, i.e.:

$$e_{pot}^{(n)} = (qE/m)\,\mathrm{y}^{(n)} \quad \text{and} \quad e_{kin}^{(n-1/2)} = \left(\mathrm{v}^{(n-1/2)}\right)^2 / 2 \tag{35}$$

Therefore to evaluate the total energy at a specific time (e.g. $t^{(n)}$), one must interpolate one of the variables to that time level. Both kinetic and potential energies are shown as function of time for the Boris algorithm in Figure 6.

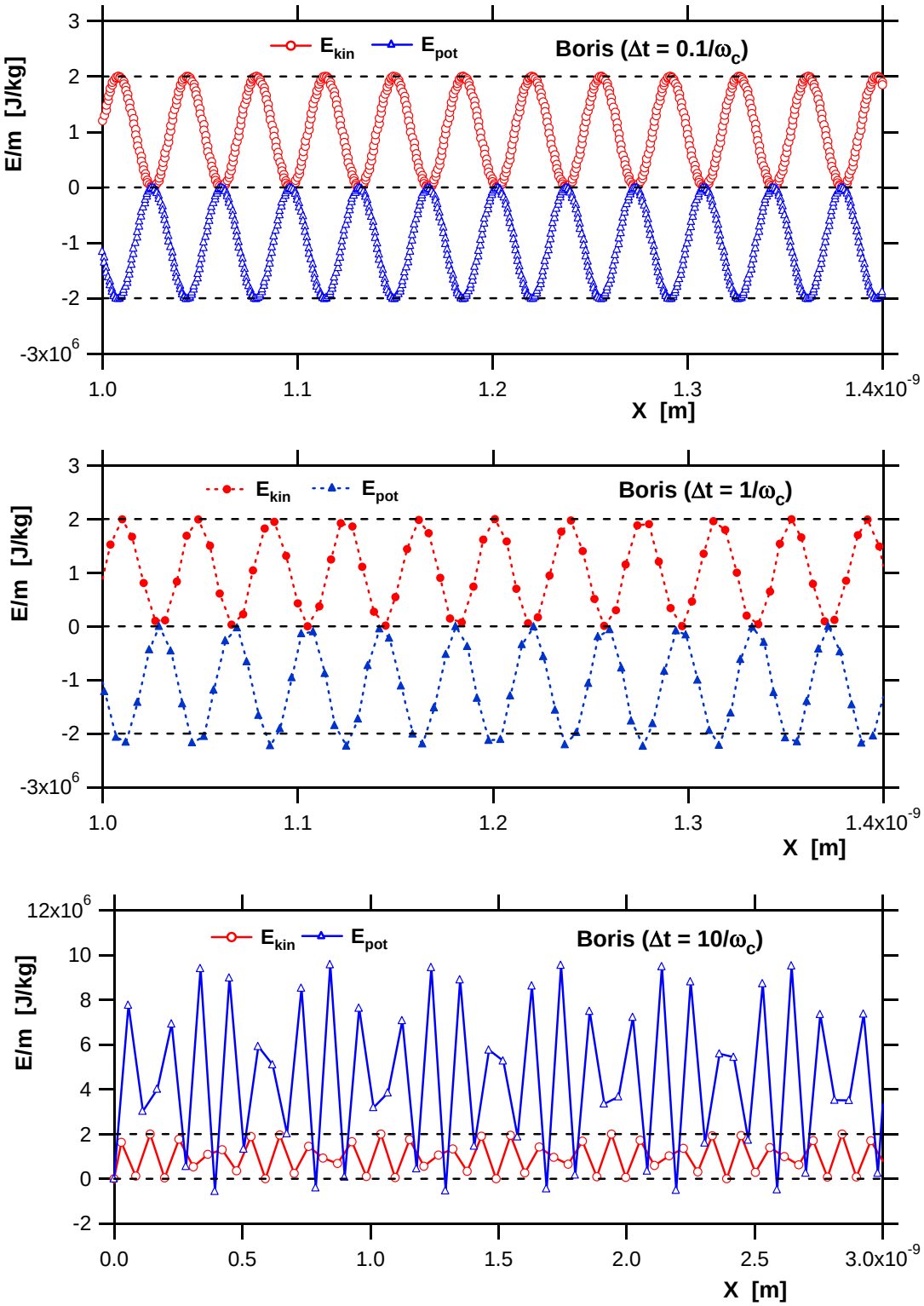

Figure 6: Kinetic and potential energy versus distance for Boris algorithm – all cases of time steps. Dashed horizontal lines indicate theoretical limits of variation.

It can be seen that there is a rapid degradation of the energy conservation as the time step is increased to values of the same order or beyond the gyro-period; the error in amplitude of the

particle trajectory leads to errors in potential energy which become severe for large time steps. Since the current method is in perfect agreement with the exact solution at all time steps, the energy is perfectly conserved in that case (see Figure 7).

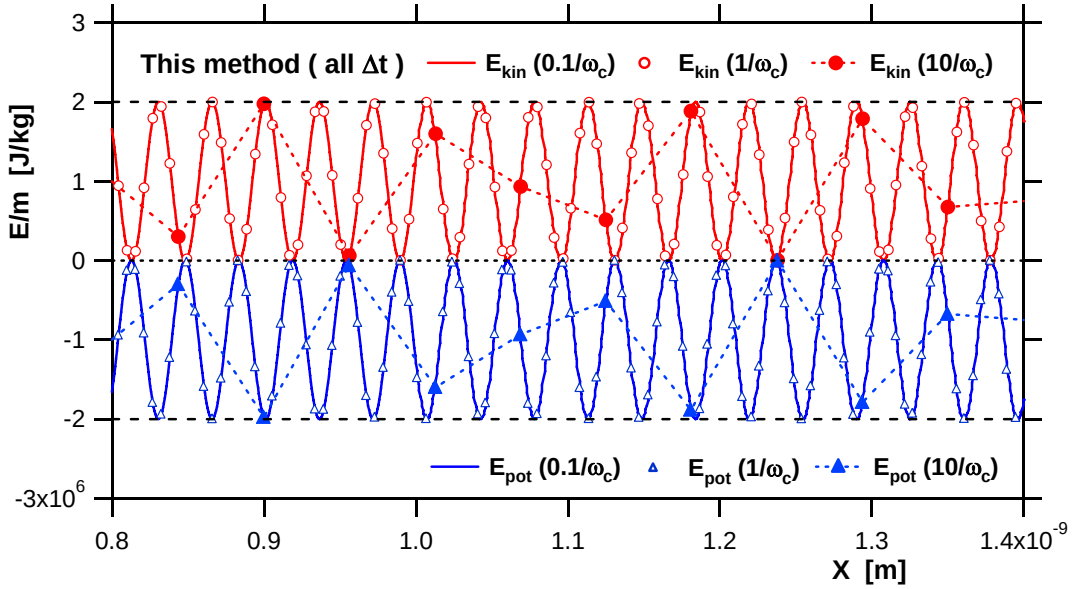

Figure 7: Kinetic and potential energy versus distance for current method – all time steps shown.

To evaluate the total energy, one must account for the dephasing of the velocity and position in the case of the Boris algorithm, as mentioned previously. This dephasing can be clearly seen when plotting both energies versus a single time coordinate ($t^{(n)}$) in Figure 8. The shift of the two curves is a result of the leap-frog algorithm. One can correct for this by plotting the kinetic energy versus the proper time of evaluation, i.e. the set $\{t^{(n+1/2)}\}$, as done in Figure 7; this shifts all the points to the left, as indicated by the black arrows of Figure 8. The total energy can be evaluated at the set of times $\{t^{(n)}\}$ by adding the potential energy at that time with the average of the kinetic energies at times $(t^{(n-1/2)}, t^{(n+1/2)})$. This interpolation is accurate as long as the time step is sufficiently small, i.e. such a linear approximation of the trajectory between the two times is reasonable; however, it can lead to severe errors for large time steps. This phase error of the Boris scheme is <u>in addition</u> to the amplitude error (effective Larmor radius) already observed, for example, in Figure 6. The average errors on total energy and position can be obtained for various values of the time step and magnetic field. The error on the total energy can be defined here as

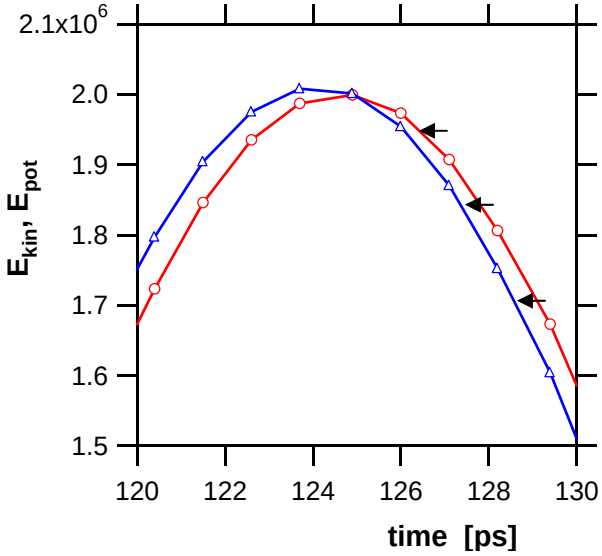

Figure 8: Kinetic (red) and potential (blue) energies versus a single time coordinate.

$err(E_{tot}) = (E_{tot} - E_{in})/E_0$, where E_{in} is the initial total energy (at $t = 0$) and E_0 is a representative energy scale; here, $E_0 = \mathrm{v}_D^2/2$, where v_D is the drift velocity. Similarly for the position, the error is defined as $err(\vec{X}) = \Sigma \left| x_\alpha - x_\alpha^{exact} \right| / X_0$; all components of the position are contributing, and X_0 is a representative length scale – in the case of cold particles here, the Larmor radius, i.e. half the height of the cycloid motion. Both errors are shown in Figure 9 for the two schemes.

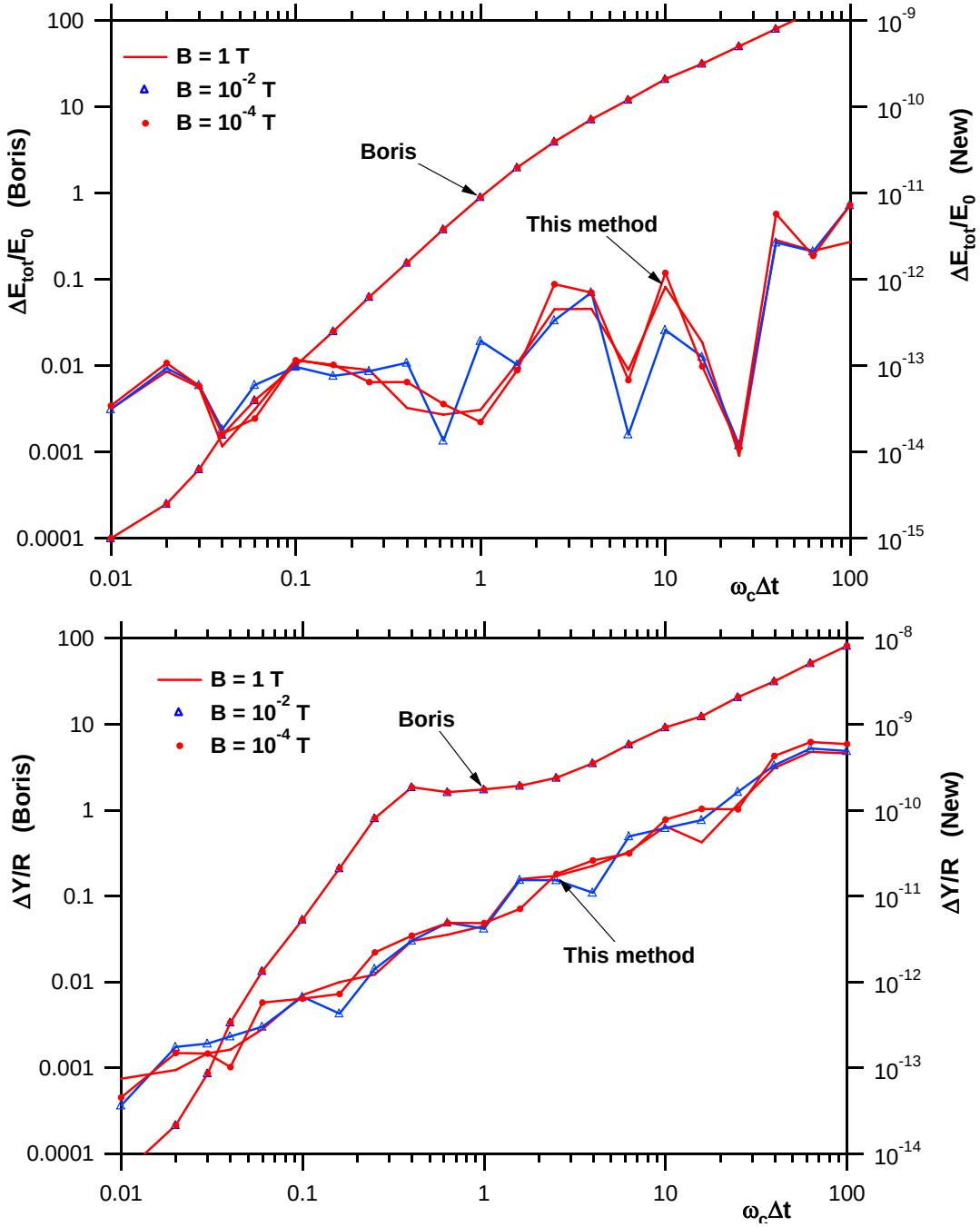

Figure 9: Error on energy (a) and position (b) as function of time step for several values of the magnetic field, for the Boris and our algorithms.

Note the change of scale between the left (Boris algorithm) and right axis (current algorithm); clearly, the present scheme is more accurate by several orders of magnitude.

5. Conclusions

We have successfully implemented and tested a new particle pusher algorithm that can effectively be used for large time steps, much larger than the gyroperiod; the method is based on the exact solution of the equations of motion, but does not require tracking the phase of the particle motion around the field line. The method can be applied to arbitrarily large time steps and yields exact (down to machine accuracy) conservation of energy and exact position. The method is currently restricted to the non-relativistic case, and to uniform fields. Extension to the relativistic regime would be very difficult, since there is no longer an analytical solution; extension to the non-uniform (magnetic) field does not present a-priori any difficulties, but this must be verified.

It would a-priori appear that the algorithm is computationally expensive, but this is not necessarily the case. The push matrices (eqs. 32) need to be computed only once for each time-step, but are the same for each particle in this case of uniform field. Thus, the method would be efficient when computing a large number of particles in such configurations, e.g. Penning traps. In the case of weakly non-uniform fields, one can also attempt a perturbation expansion, such that the computationally expensive push matrices (involving trigonometric function evaluations) are again computed once in each computational cell, while each particle is transported according to a hybrid scheme involving the one described here, and a rapid scheme such as the Boris algorithm for the small perturbation $\delta\vec{B}$ or a similar procedure that does not involve time staggering. This can be investigated in the future.

6. References

[1] J. P. Boris, "Relativistic plasma simulation – optimization of a hybrid code", *Proc. 4th Conf. Numerical. Simulation of Plasmas*, pp. 3-67, J.P. Boris and R.A. Shanny eds, Naval Research Laboratory, Wash. D.C., Nov. 2-3 1970.

[2] C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation*, pp. 58-63, Inst. of Phys. Publ., Bristol, UK, 1991.

[3] W. W. Lee, *Phys. Fluids* **26**(2), 556-562 (1983).

Appendix A: Boris Algorithm

The Boris algorithm is defined by the following steps, from the velocity at $t - \Delta t/2$ and the fields at t:

1. $\mathbf{v}^- = \mathbf{v}(t-\Delta t/2) + \frac{q}{m}\frac{\Delta t}{2}\mathbf{E}(t)$ (A.1)

2. $\mathbf{v}' = \mathbf{v}^- + \frac{q}{m}\frac{\Delta t}{2}\mathbf{v}^- \times \mathbf{B}(t)$ (A.2)

3. $\mathbf{v}^+ = \mathbf{v}^- + \frac{\frac{q}{m}\frac{\Delta t}{2}}{1+\left(\frac{q}{m}\frac{\Delta t}{2}|\mathbf{B}|\right)^2}\mathbf{v}' \times \mathbf{B}(t)$ (A.3)

4. $\mathbf{v}(t+\Delta t/2) = \mathbf{v}^+ + \frac{q}{m}\frac{\Delta t}{2}\mathbf{E}(t)$ (A.4)

The position is advanced by the additional step:

5. $\mathbf{r}(t+\Delta t) = \mathbf{r}(t) + \Delta t\, \mathbf{v}(t+\Delta t/2)$ (A.5)

The algorithm could also be written a different way. Let us define the following vector:

$$\boldsymbol{\beta} = \frac{\omega_c \Delta t}{2}\hat{\mathbf{b}} \tag{A.6}$$

where $\omega_c = \frac{q}{m}|\mathbf{B}|$ is the cyclotron frequency (unsigned) and $\hat{\mathbf{b}} = \mathbf{B}/|\mathbf{B}|$ is the unit vector along the magnetic field. Steps 2 and 3 can be combined into the form:

$$\mathbf{v}^+ = \mathbf{v}^- - \frac{2}{1+\beta^2}\boldsymbol{\beta}\times\mathbf{v}^- + \frac{2}{1+\beta^2}\boldsymbol{\beta}\times\boldsymbol{\beta}\times\mathbf{v}^- \tag{A.7}$$

(with $\beta \equiv |\boldsymbol{\beta}|$). The equivalent matrix form is:

$$\mathbf{v}^+ = \mathbf{v}^- + \frac{2}{1+\beta^2}\begin{pmatrix} 0 & +\beta_z & -\beta_y \\ -\beta_z & 0 & +\beta_x \\ +\beta_y & -\beta_x & 0 \end{pmatrix}\cdot\mathbf{v}^- + \frac{2}{1+\beta^2}\begin{pmatrix} -\beta_y^2-\beta_z^2 & \beta_x\beta_y & \beta_x\beta_z \\ \beta_y\beta_x & -\beta_x^2-\beta_z^2 & \beta_y\beta_z \\ \beta_z\beta_x & \beta_z\beta_y & -\beta_x^2-\beta_y^2 \end{pmatrix}\cdot\mathbf{v}^- \tag{A.8}$$

or equivalently:

$$\mathbf{v}^+ = \frac{1-\beta^2}{1+\beta^2}\mathbf{v}^- + \frac{2}{1+\beta^2}\begin{pmatrix} 0 & +\beta_z & -\beta_y \\ -\beta_z & 0 & +\beta_x \\ +\beta_y & -\beta_x & 0 \end{pmatrix}\cdot\mathbf{v}^- + \frac{2}{1+\beta^2}\begin{pmatrix} \beta_x\beta_x & \beta_x\beta_y & \beta_x\beta_z \\ \beta_y\beta_x & \beta_y\beta_y & \beta_y\beta_z \\ \beta_z\beta_x & \beta_z\beta_y & \beta_z\beta_z \end{pmatrix}\cdot\mathbf{v}^- \tag{A.9}$$

The origin of the Boris algorithm is made clear by the following. Consider the rotation step as follows:

$$\delta\mathbf{v} = \mathbf{v}^+ - \mathbf{v}^- = \frac{q}{m}\frac{\Delta t}{2}(\mathbf{v}^- + \delta\mathbf{v})\times\mathbf{B}(t) \tag{A.10}$$

leading to:

$$\mathbf{v}^+ + \frac{q}{m}\frac{\Delta t}{2}\mathbf{B}\times\mathbf{v}^+ = \mathbf{v}^- - \frac{q}{m}\frac{\Delta t}{2}\mathbf{B}\times\mathbf{v}^- \tag{A.11}$$

or in matrix form:

$$\underbrace{\begin{pmatrix} 1 & -\beta_z & +\beta_y \\ +\beta_z & 1 & -\beta_x \\ -\beta_y & +\beta_x & 1 \end{pmatrix}}_{\mathbf{N}_+} \cdot \mathbf{v}^+ = \underbrace{\begin{pmatrix} 1 & +\beta_z & -\beta_y \\ -\beta_z & 1 & +\beta_x \\ +\beta_y & -\beta_x & 1 \end{pmatrix}}_{\mathbf{N}_-} \cdot \mathbf{v}^- \tag{A.12}$$

The matrix $\mathbf{N}_+$ on the LHS can be inverted to yield:

$$\mathbf{N}_+^{-1} = \frac{1}{1+\beta^2} \begin{pmatrix} 1+\beta_x^2 & +\beta_z+\beta_x\beta_y & -\beta_y+\beta_x\beta_z \\ -\beta_z+\beta_y\beta_x & 1+\beta_y^2 & +\beta_x+\beta_y\beta_z \\ +\beta_y+\beta_z\beta_x & -\beta_x+\beta_z\beta_y & 1+\beta_z^2 \end{pmatrix} \tag{A.13}$$

The product $\mathbf{N}_+^{-1} \cdot \mathbf{N}_-$ is:

$$\mathbf{N}_+^{-1} \cdot \mathbf{N}_- = \frac{1}{1+\beta^2} \begin{pmatrix} 1+\beta_x^2-\beta_y^2-\beta_z^2 & +2\beta_z+2\beta_x\beta_y & -2\beta_y+2\beta_x\beta_z \\ -2\beta_z+2\beta_y\beta_x & 1+\beta_y^2-\beta_x^2-\beta_z^2 & +2\beta_x+2\beta_y\beta_z \\ +2\beta_y+2\beta_z\beta_x & -2\beta_x+2\beta_z\beta_y & 1+\beta_z^2-\beta_x^2-\beta_y^2 \end{pmatrix} \tag{A.14}$$

which can be decomposed into the form:

$$\mathbf{N}_+^{-1} \cdot \mathbf{N}_- = \frac{1-\beta^2}{1+\beta^2}(\mathbf{1}) + \frac{2}{1+\beta^2} \begin{pmatrix} 0 & +\beta_z & -\beta_y \\ -\beta_z & 0 & +\beta_x \\ +\beta_y & -\beta_x & 0 \end{pmatrix} + \frac{2}{1+\beta^2} \begin{pmatrix} \beta_x\beta_x & \beta_x\beta_y & \beta_x\beta_z \\ \beta_y\beta_x & \beta_y\beta_y & \beta_y\beta_z \\ \beta_z\beta_x & \beta_z\beta_y & \beta_z\beta_z \end{pmatrix} \tag{A.15}$$

We see that this is equivalent to (A.9), and therefore the steps 2 and 3 of the Boris algorithm are equivalent to a time-centered scheme for the gyro-motion (A.10).

Note that the Boris algorithm is operator-splitting the electric acceleration from the magnetic rotation. We could also look into a complete operator definition without this splitting, by considering the full time-centered algorithm:

$$\delta\mathbf{v} = \mathbf{v}_{n+1/2} - \mathbf{v}_{n-1/2} = \frac{q}{m}\Delta t \cdot \mathbf{E}_n + \frac{q}{m}\frac{\Delta t}{2}(\mathbf{v}_{n+1/2} + \mathbf{v}_{n-1/2}) \times \mathbf{B}_n \tag{A.16}$$

which becomes

$$\mathbf{N}_+ \cdot \mathbf{v}_{n+1/2} = \mathbf{a}_n \, \Delta t + \mathbf{N}_- \cdot \mathbf{v}_{n-1/2} \tag{A.17}$$

where $\mathbf{a}_n = (q/m) \cdot \mathbf{E}_n$. The solution is already expressed using the matrices of (A.13) and (A.15). To simplify the notation, let us define the following:

$$\mathbf{M}_0 = \frac{1}{1+\beta^2}(\mathbf{1}), \quad \mathbf{M}_1 = \frac{1}{1+\beta^2} \begin{pmatrix} 0 & +\beta_z & -\beta_y \\ -\beta_z & 0 & +\beta_x \\ +\beta_y & -\beta_x & 0 \end{pmatrix} \text{ and } \mathbf{M}_2 = \frac{1}{1+\beta^2} \begin{pmatrix} \beta_x\beta_x & \beta_x\beta_y & \beta_x\beta_z \\ \beta_y\beta_x & \beta_y\beta_y & \beta_y\beta_z \\ \beta_z\beta_x & \beta_z\beta_y & \beta_z\beta_z \end{pmatrix} \tag{A.18}$$

then the solution is expressed as:

$$\mathbf{v}_{n+1/2} = \mathbf{N}_+^{-1} \cdot (\mathbf{a}_n \Delta t) + (\mathbf{N}_+^{-1}\mathbf{N}_-) \cdot \mathbf{v}_{n-1/2} \tag{A.19}$$

with:

$$\mathbf{N}_+^{-1} = \mathbf{M}_0 + \mathbf{M}_1 + \mathbf{M}_2 \tag{A.20a}$$

and

$$\mathbf{N}_+^{-1}\mathbf{N}_- = (1-\beta^2)\mathbf{M}_0 + 2\,\mathbf{M}_1 + 2\,\mathbf{M}_2 \tag{A.20b}$$

In the case where the magnetic field is aligned along the $\hat{z}$-axis, considerable simplification occurs. We can easily see that (A.9) leads to the following relations for the parallel and transverse components respectively:

$$\mathbf{v}_\parallel^+ \equiv \mathbf{v}_\parallel^- \tag{A.21a}$$

and $$\mathbf{v}_{\perp}^{+} - \mathbf{v}_{\perp}^{-} = \frac{1}{1+(\delta/2)^2}\begin{pmatrix} -\delta^2/2 & \delta \\ -\delta & -\delta^2/2 \end{pmatrix} \cdot \mathbf{v}_{\perp}^{-} \qquad \text{(A.22b)}$$

where $\delta = \omega_c \Delta t$. This leads to:

$$\begin{aligned} \Delta\mathbf{v} &= \mathbf{v}_{n+1/2} - \mathbf{v}_{n-1/2} \\ &= (\cdot\mathbf{v}^{+} + \tfrac{1}{2}\mathbf{a}_n \Delta t) - (\cdot\mathbf{v}^{-} - \tfrac{1}{2}\mathbf{a}_n \Delta t) \qquad \text{(A.23a)} \\ &= \mathbf{a}_n \Delta t + \frac{1}{1+(\delta/2)^2}\begin{pmatrix} -\delta^2/2 & \delta & 0 \\ -\delta & -\delta^2/2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot (\mathbf{v}_n + \tfrac{1}{2}\mathbf{a}_n \Delta t) \end{aligned}$$

or, keeping terms of order δ^2 only:

$$\Delta\mathbf{v} = \begin{cases} \mathrm{a}_x \Delta t + (\omega_c \Delta t)(\mathrm{v}_y + \tfrac{1}{2}\mathrm{a}_y \Delta t) - \tfrac{1}{2}(\omega_c \Delta t)^2 \mathrm{v}_x \\ \mathrm{a}_y \Delta t - (\omega_c \Delta t)(\mathrm{v}_x + \tfrac{1}{2}\mathrm{a}_x \Delta t) - \tfrac{1}{2}(\omega_c \Delta t)^2 \mathrm{v}_y \\ \mathrm{a}_z \Delta t \end{cases} \qquad \text{(A.23b)}$$